<center>**Remarks**</center>

Applicant respectfully requests reconsideration of this application as amended.

Claims 1, 3, 13, 17, and 27-36 have been amended. Claims 2, 14, and 18 have been

cancelled. Claims 37-39 have been added. Therefore, claims 1, 3-7, 13, 15-17, 19-21, and

27-39 are presented for examination.

<center>**35 U.S.C. §103(a) Rejection**</center>

Claims 1-7, 13-21 and 27-36 stand rejected under 35 U.S.C. §103(a) as being

unpatentable over Archambault (U.S. Patent No. 6,173,444) in view of Blainey (U.S. Patent

No. 6,045,585) in view of "Restricted Pointers are Coming" by Robison. Applicant submits

that the present claims are patentable over Archambault and Blainey, in view of Robison.

Archambault discloses a method that reduces the size of alias sets associated with

program pointers through the use of a pointer alias graph. Standard data flow gathering

techniques are used to develop the pointer alias graph. The nodes in the graph represent

either a definition of a pointer variable or a use of a pointer variable, and each node has an

associated alias set. The initial alias sets for definition nodes is the right hand side of the

pointer variable assignment operation, and the initial alias set for use nodes is the value of the

object at that execution point. Location information, the basic block number (relative to the

flow graph) and position within the basic block, is saved for each node. (Archambault at col.

5, ll. 4-17).

Claim 1, as amended, recites:

> A method comprising:
>   receiving a code segment having a plurality of
> instructions, the code segment having an outer scope and
> a number of inner scopes, wherein the plurality of

instructions comprise a number of pointers, wherein at least one of the number of pointers is a restricted pointer;

    determining a base pointer for each pointer of the number of pointers;

    determining a scope of the at least one restricted pointer relative to scopes of each of the number of pointers in the code segment; and

    <u>determining whether at least one pointer of the number of pointers is aliased with the at least one restricted pointer, based on the base pointer and scope of each of the restricted pointer and the at least one pointer, when the at least one restricted pointer is out-of-scope relative to the at least one pointer.</u>

Applicant submits that Archambault does not disclose or suggest <u>determining whether a pointer is aliased with a restricted pointer, based on the base pointer and scope of each of the restricted pointer and the pointer, when the restricted pointer is out-of-scope relative to the pointer.</u> First, Archambault does not disclose *using base pointers* to determine whether an alias exists with a restricted pointer. The Office Action cites column 5, lines 4-17 of Archambault as disclosing the use of base pointers. (Office Action dated 1/7/05, at pg. 7.) This portion of Archambault discusses an "alias set" for nodes of a pointer alias graph. The nodes in the graph represent either a definition of a pointer variable or a use of a pointer variable. The alias set for definition nodes is the "right hand side of the pointer variable assignment operation" and the alias set for use nodes is the "value of the object at that execution point." (Archambault at col. 5., ll. 10-15.) However, applicant submits that this "alias set" is not the same as a base pointer for a pointer. As disclosed in Archambault, the alias set is for a node, not a pointer. Furthermore, the alias set is either an assignment operation or a value of an object at an execution point, not another pointer.

Second, Archambault does not disclose *using scope* to determine whether an alias exists with a restricted pointer. The only reference to Archambault disclosing the utilization

of scope in the Office Action is with respect to claim 5, where column 5 lines 52-56 of Archambault are cited as disclosing "determining the pointer variables accessed in the local scope." (Office Action at pg. 8.) Applicant submits that this cited portion does not disclosing determining the scope of a restricted pointer in relation to the scope of another pointer, and using this scope information to determine whether an alias exists between the restricted pointer and the other pointer. Furthermore, applicant can find no disclosure elsewhere in Archambault of such a feature. Therefore, the features of claim 1 are not disclosed or suggested by Archambault.

Blainey discloses a system and method for determining alias information at an inter-compilation unit level of a compilation process. (Blainey at Abstract.) Robison discloses the use of restricted pointers in C and C++. (Robison at pg. 1.) Applicant submits that neither Blainey nor Robison disclose or suggest determining whether a pointer is aliased with a restricted pointer, based on the base pointer and scope of each of the restricted pointer and the pointer, when the restricted pointer is out-of-scope relative to the pointer. Therefore, the features of claim 1 are not disclosed or suggested by Blainey or Robison.

As a result, Archambault, Blainey, and Robison, individually or in combination, do not disclose or suggest the features of claim 1. Therefore, claim 1 is patentable over Archambault and Blainey, in view of Robison. Claims 3-7 and 27-31 depend from claim 1 and include additional limitations. As such, claims 3-7 and 27-31 are also patentable over Archambault and Blainey, in view of Robison.

Independent claims 13 and 17 contain similar features to claim 1, such as determining whether a pointer is aliased with a restricted pointer, based on the base pointer and scope of each of the restricted pointer and the pointer, when the restricted pointer is out-of-scope

relative to the pointer. As discussed above, Archambault, Blainey, and Robison, individually or in combination, do not disclose or suggest such a feature. Therefore, claims 13 and 17 are also patentable over Archambault and Blainey, in view of Robison, for the reasons discussed with respect to claim 1.

Claims 15, 16, and 37-39 depend from claim 13 and include additional limitations. Claims 19-21 and 32-36 depend from claim 17 and include additional limitations. Therefore, claims 15, 16, 19-21, and 32-39, are also patentable over Archambault and Blainey, in view of Robison.

Applicant respectfully submits that the rejections have been overcome and that the claims are in condition for allowance. Accordingly, applicant respectfully requests the rejections be withdrawn and the claims be allowed.

The Examiner is requested to call the undersigned at (303) 740-1980 if there remains any issue with allowance of the case.
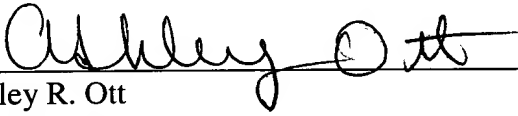
Applicant respectfully petitions for an extension of time to respond to the outstanding Office Action pursuant to 37 C.F.R. § 1.136(a) should one be necessary. Please charge our Deposit Account No. 02-2666 to cover the necessary fee under 37 C.F.R. § 1.17(a) for such an extension.

Please charge any shortage to our Deposit Account No. 02-2666.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: <u>April 6, 2005</u>

Ashley R. Ott
Reg. No. 55,515

12400 Wilshire Boulevard
7th Floor
Los Angeles, California 90025-1026
(303) 740-1980